

# 分段式位隙分组帧时隙 Aloha 算法

邢志鹏, 杨恒新, 张 昀

(南京邮电大学 电子科学与工程学院, 江苏 南京 210003)

**摘 要:**自从射频识别(RFID)技术越来越频繁地应用于各种场合,提高 RFID 标签的识别效率就成为了一个重要任务。文中在 FSA 的基础上,结合位隙分组分析,提出分段式位隙分组帧时隙 Aloha 算法。该算法将位隙分为两段,在标签发生碰撞时,由该时隙内发生碰撞的标签先后发送前后两段位隙标志位进行再识别;若在前段位隙中成功识别该时隙内所有碰撞的标签,则进入下一个时隙的标签识别;若经过前段位隙再识别后仍存在碰撞标签,则继续发送后段位隙进行识别。理论分析表明,当位隙分为等长两部分时,系统吞吐率曲线在一定标签数量(文中取 10 000)范围内可以保持平坦,在吞吐率达到 65% 以上的同时,系统通信量达到最低。

**关键词:**射频识别;防碰撞算法;Aloha;位隙分组

**中图分类号:**TP301.6

**文献标识码:**A

**文章编号:**1673-629X(2016)04-0031-05

doi:10.3969/j.issn.1673-629X.2016.04.007

## A Frame Slot Aloha Algorithm of Sectional Bit-slot Group

XING Zhi-peng, YANG Heng-xin, ZHANG Yun

(College of Electronic Science and Engineering, Nanjing University of Posts and Telecommunications,  
Nanjing 210003, China)

**Abstract:** Since the radio frequency identification (RFID) technology is more and more frequently used in various occasions, it is an important task to improve the recognition efficiency of RFID tags. Combined with bit-slot grouping, a sectional bit-slot group frame slot Aloha algorithm is proposed in this paper based on the classical FSA. The algorithm divides the bit-slot into two sections, when collision occurs to a particular slot, the collided tags send the forward bit-slot and backward bit-slot to the reader for recognition; if the reader identifies all the collided tags successfully in the forward bit-slot, the reader enters the next slot of tag identification. If the collision still exists after the recognition in the forward bit-slot, the collided tags continue to send backward bit-slot to the reader for recognition. Theoretical analysis shows when a bit-slot is divided into two isometric parts, system throughput curve maintains flat at the range of a certain number of tags (within 10 000). Furthermore, while the throughput is over 65%, the system communication traffic reaches the lowest.

**Key words:** RFID; anti-collision algorithm; Aloha; bit-slot group

## 0 引言

作为物联网的关键技术之一,RFID 是一种利用空间耦合或电磁感应传递射频信号进行无接触的自动识别技术<sup>[1-4]</sup>。早在 20 世纪 70 年代,大量的研究人员就开始投身于 RFID 技术的研究。进入 21 世纪后,随着物联网概念的提出,RFID 得到了更多的关注并广泛应用于服务业、制造业等行业。

由于 RFID 标签可以无需视距地从远距离(比如 2~3 m)被高速阅读(比如每秒 100 标签),因此 RFID 技术在众多应用中变得越来越普遍。在大多数情况

下,一个 RFID 系统由一系列阅读器和电子标签组成。阅读器通常是一个具有足够内存和读写区域的有源设备。标签用于标识物件,可分为有源标签、半有源标签和无源标签。其中,无源标签具有工艺简单、价格低廉、使用寿命长等诸多优点,实际应用规模较大。无源标签从阅读器的电磁场中获取数据传输所需要的能量<sup>[5]</sup>。RFID 系统正常运行时,阅读器查询它阅读区域内的电子标签,同时激活电子标签内部的射频兼容电路来产生足够的能量进行必要的操作和数据传输。电子标签使用这些能量可以生成回送给阅读器的应答。

收稿日期:2015-07-20

修回日期:2015-10-22

网络出版时间:2016-03-22

基金项目:国家自然科学基金青年科学基金项目(61302155)

作者简介:邢志鹏(1989-),男,硕士研究生,研究方向为智能信息处理;杨恒新,副教授,研究方向为无线射频识别技术;张 昀,硕士生导师,研究方向为通信信号盲检测、神经网络和无线传感器网络等。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20160322.1522.088.html>

一旦阅读器获取到电子标签回送过来的应答,就立即从后端服务器(或者数据库)中检索对应标签的信息<sup>[6]</sup>。

在标签应答收集阶段,阅读器通过共享的无线通信信道识别每一个电子标签。当阅读器查询区域内的多个电子标签同时向阅读器应答时,信号会互相碰撞<sup>[7]</sup>,阅读器将无法识别合法标签。这种现象干扰了阅读器对标签的识别过程,浪费了更多的带宽,花费了更高的通信开销,而且提高了标签识别过程中的传输时延。因此,一个有效的标签信号碰撞仲裁机制是快速识别 RFID 系统中不可或缺的<sup>[8]</sup>。

传统的基于 Aloha 的概率型算法存在的缺陷:由于阅读器发出的帧长有限,而多标签响应具有很大的随机性,因此,可能导致某一标签在相当长的一段时间内无法被识别。而且,当查询范围内标签数量继续增大时,系统性能会急剧下降、能耗大、出现“标签饥饿”现象。改进型 Aloha 算法大多将查询范围内的标签分组,从而解决一帧中标签数量大的问题。而标签分组本身就存在随机性,且反复地对标签进行分组增加了指令开销,附加了额外的通信量<sup>[9]</sup>。

在研究经典 Aloha 以及相关改进算法的基础上,文中提出一种分段式位隙分组帧时隙 Aloha 算法。该算法通过分段的位隙标志位对一个碰撞时隙内的标签进行再识别,从而有效避免了多余时隙的传送,在保持高吞吐率的同时减少了系统的通信总量。

## 1 经典标签防碰撞算法及其改进算法

现存 RFID 系统的防碰撞算法主要分为两类:标签防碰撞算法和阅读器防碰撞算法。标签防碰撞算法又可分成两类:确定型算法和概率型算法。

确定型算法又称为树型算法,主要是基于标签 ID 的搜索协议:阅读器通过改变查询命令,使标签基于 ID 号逐级分组,直到有唯一的标签被阅读器识别为止。基于树的防碰撞算法主要包括 TS(Tree Splitting)、BT(Binary Tree)、BS(Binary Search)、QT(Query Tree)、CT(Collision Tree),以及在 CT 上改进的 ICT(Improved Collision Tree)。

传统的概率型 RFID 防碰撞算法包括 PA(Pure Aloha)、SA(Slotted Aloha)、FSA(Frame Slotted Aloha)。其中 FSA 又衍生出 BFSA(Basic Frame Slotted Aloha)、DFSA(Dynamic Frame Slotted Aloha)以及 EDFSA(Enhanced Dynamic Frame Slotted Aloha)<sup>[10]</sup>。

FSA 算法把若干个时隙分为一帧,在一轮查询中,标签随机选择帧内时隙对阅读器做出响应。当两个以上的标签选择了同一时隙时,就会发生碰撞。发生碰撞的标签只得等到下一帧进行识别。

文献[11]中提出了一种位隙分组帧时隙算法(BGFSA)。该算法把一定长度的位隙标志位(设为  $2L$  位)置于标签 ID 的后面,标签响应阅读器时随机选取位隙标志位中一位为 1,其余全为 0,在发生碰撞时,阅读器解析出置 1 位隙标志位的位置,然后从高到低依次把位置信息回传给标签,冲突时隙内的标签对该位置信息进行匹配,匹配成功的标签再向阅读器应答。理论上,在一个冲突时隙内,最多可识别出  $2L$  个冲突的标签。

## 2 位隙分段

假若把位隙分段,则在一个冲突时隙内能够识别的标签个数就会大于  $2L$ 。在最优情况下,把  $2L$  位位隙标志位分为等长的两段,则一个冲突时隙内最多能够识别  $L^2$  个标签。假如  $2L=8$ ,则分段后,最多能够识别出 16 个标签。

先考虑固定帧长的情况。设帧长为  $N$ ,查询范围内标签数为  $n$ ,位隙标志位长度为  $2L$ ,即每段长度均为  $L$ 。

那么,单标签时隙的概率为:

$$P_{\text{succ}} = C_n^1 * \left(\frac{1}{N}\right) * \left(1 - \frac{1}{N}\right)^{n-1} \quad (1)$$

空闲时隙的概率为:

$$P_{\text{idle}} = C_n^0 * \left(1 - \frac{1}{N}\right)^n = \left(1 - \frac{1}{N}\right)^n \quad (2)$$

冲突时隙的概率为:

$$P_{\text{coll}} = 1 - P_{\text{succ}} - P_{\text{idle}} = 1 - \left(1 - \frac{1}{N}\right)^n - C_n^1 * \left(\frac{1}{N}\right) * \left(1 - \frac{1}{N}\right)^{n-1} \quad (3)$$

一个查询周期内的冲突时隙数为:

$$N_{\text{coll}} = N * P_{\text{coll}} \quad (4)$$

同样,成功识别的标签数为:

$$N_{\text{succ}} = N * P_{\text{succ}} \quad (5)$$

一个查询周期内平均每个冲突时隙的标签数为:

$$t = \frac{n - N_{\text{succ}}}{N_{\text{coll}}} \quad (6)$$

位隙标志位分段后,由于前后两段等长且随机在自己的段中置 1,因此位隙等效长度为  $L^2$ ,则等效时隙长度为:

$$N_{\text{eq}} = N * L^2 \quad (7)$$

等效时隙下单标签时隙的概率为:

$$P_{\text{eq-succ}} = C_n^1 * \left(\frac{1}{N_{\text{eq}}}\right) * \left(1 - \frac{1}{N_{\text{eq}}}\right)^{n-1} \quad (8)$$

因此一个查询周期内总的成功识别的标签数为:

$$N_{\text{read}} = N_{\text{eq}} * P_{\text{eq-succ}} = n * \left(1 - \frac{1}{N * L^2}\right)^{n-1} \quad (9)$$

由于在冲突时隙内使用位隙进行再识别,等效于增加了额外的时隙。前段位隙增加的额外时隙为:

$$N_F = L * (1 - P_{FBS-idle}) \tag{10}$$

其中

$$P_{FBS-idle} = \left(1 - \frac{1}{L}\right)^t \tag{11}$$

通过前段位隙进行再识别后,还剩下的标签数为:

$$t_{remain} = t - L * P_{FBS-succ} \tag{12}$$

其中

$$P_{FBS-succ} = C_t^1 * \left(\frac{1}{L}\right) * \left(1 - \frac{1}{L}\right)^{t-1} \tag{13}$$

在前段位隙产生的冲突时隙数为:

$$N_{FBS-coll} = L * (1 - P_{FBS-idle} - P_{FBS-succ}) \tag{14}$$

$$\eta = \frac{N_{read}}{N + N_{extra}} =$$

$$n * \left(1 - \frac{1}{N * L^2}\right)^{n-1}$$

$$\frac{N + N * \left(1 - \left(1 - \frac{1}{N}\right)^n - C_n^1 * \left(\frac{1}{N}\right) * \left(1 - \frac{1}{N}\right)^{n-1}\right) * \left(L * \left(1 - \left(1 - \frac{1}{L}\right)^t\right) + L * \left(1 - \left(1 - \frac{1}{L}\right)^k\right) * L * \left(1 - \left(1 - \frac{1}{L}\right)^t - C_t^1 * \left(\frac{1}{L}\right) * \left(1 - \frac{1}{L}\right)^{t-1}\right)\right)}{\tag{20}}$$

其中

$$t = \frac{n - n * \left(1 - \frac{1}{N}\right)^{n-1}}{N * \left(1 - \left(1 - \frac{1}{N}\right)^n - C_n^1 * \left(\frac{1}{N}\right) * \left(1 - \frac{1}{N}\right)^{n-1}\right)} \tag{21}$$

$$k = \frac{t - L * C_t^1 * \left(\frac{1}{L}\right) * \left(1 - \frac{1}{L}\right)^{t-1}}{L * \left(1 - \left(1 - \frac{1}{L}\right)^t - C_t^1 * \left(\frac{1}{L}\right) * \left(1 - \frac{1}{L}\right)^{t-1}\right)} \tag{22}$$

取  $N = 64$ 、 $L = 8$ , 观察系统吞吐率曲线, 如图 1 所示。

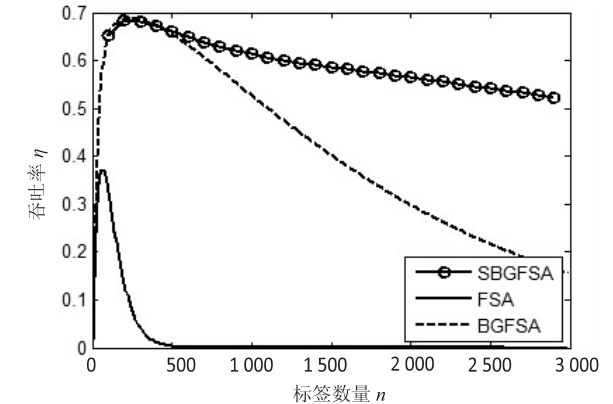


图1 三种算法吞吐率曲线

可见 SBGFSA 算法在保持了高吞吐率的同时,在标签数量较多时,吞吐率依然可以保持在 50% 以上。  
令位隙标志位长度  $2L = 24$ , 不同帧长下系统的吞

平均每个 FBS 的冲突时隙内标签数为:

$$k = \frac{t_{remain}}{N_{FBS-coll}} \tag{15}$$

后段位隙增加的额外时隙数为:

$$N_B = L * (1 - P_{BBS-idle}) * N_{FBS-coll} \tag{16}$$

其中

$$P_{BBS-idle} = \left(1 - \frac{1}{L}\right)^k \tag{17}$$

所以,一个冲突时隙内增加的额外时隙数为:

$$N_{extra-cs} = N_F + N_B \tag{18}$$

一个查询周期增加的总的额外时隙数为:

$$N_{extra} = N_{coll} * N_{extra-cs} \tag{19}$$

因此,整个系统的吞吐率为:

吐率曲线如图 2 所示。

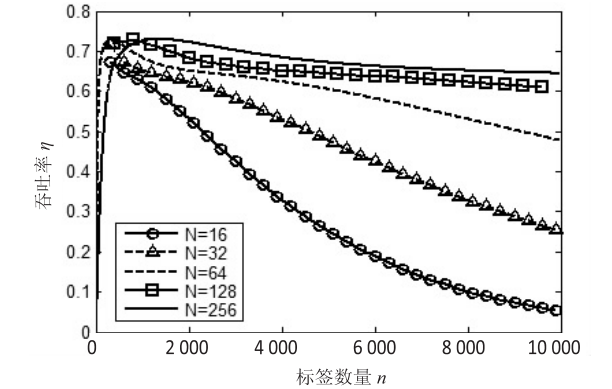


图2 SBGFSA 在位隙标志位长度为 24 时  
不同帧长下的吞吐率曲线

一般最大帧长为  $256^{[12]}$ , 这里把最大帧长就定为 256。由图 2 可见,当帧长为 256 时,在标签数量较高的情况下系统吞吐率是最大的,且在一定标签范围内,系统吞吐率近似可以看成一定值,约为 70% 左右,但在标签数量较低情况下,系统吞吐率会下降。

为提高标签数量较低情况下的系统吞吐率,在图 2 的基础上取  $N = 32$  与  $N = 256$  两条曲线,如图 3 所示。根据标签的数量,动态改变帧长。以两条曲线的交点为帧长调整点,调整方法见表 1。改进后的吞吐率曲线为图 3 中加粗实线部分。

表1 SBGFSA 算法帧长调整机制

标签数量 $n$	帧长 $N$
1 ~ 490	32
490 ~ 10 000	256

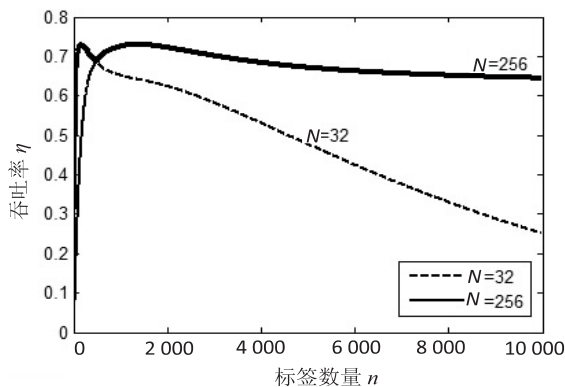
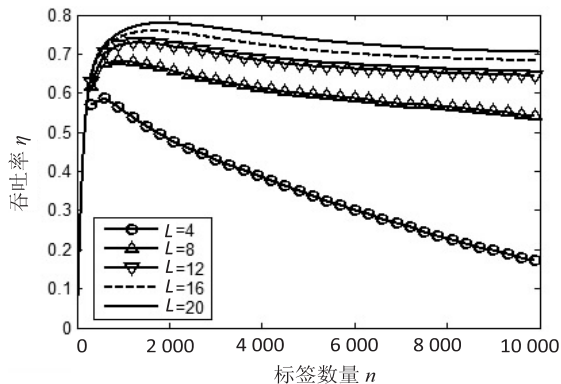


图3 SBFSA 算法动态帧长调整机制

按照如上所述,该算法能以近 70% 的信道利用率识别一万以内的标签。

### 3 最优位隙长度

为研究吞吐率与位隙长度之间的关系,固定帧长  $N=256$ , 改变位隙长度  $2L$ 。吞吐率曲线如图 4 所示。

图4 SBFSA 算法在帧长为 256 时  
不同位隙长度下的吞吐率曲线

由图 4 可见,在  $L$  达到 12 以后,随着  $L$  的增加,曲线变化越来越小。

位隙标志位的引入会提高吞吐率,但同时也增加了阅读器与标签的通信量,最优位隙长度的选取应兼顾通信量。

由式(20)可得,平均识别一个标签所需要的时隙数为:

$$N_{\text{eachTag}} = \frac{N + N_{\text{extra}}}{N_{\text{read}}} = \frac{1}{\eta} \quad (23)$$

识别所有标签需要的总时隙数为:

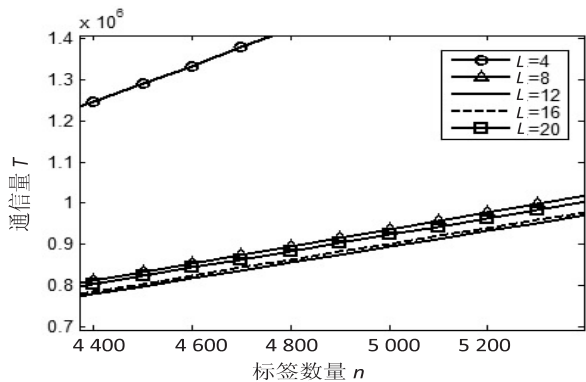
$$N_{\text{allTag}} = n * N_{\text{eachTag}} = \frac{n}{\eta}, n \in (0, 10\ 000] \quad (24)$$

设标签 ID 信息长 96 位,即  $L_{\text{TagID}} = 96$ 。则通信量(此处忽略不同算法下阅读器查询命令长度差异)为:

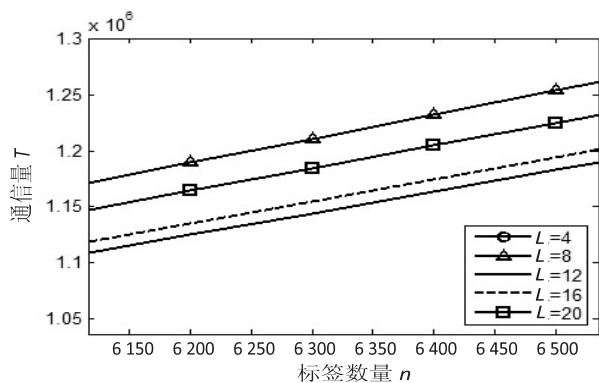
$$\text{Traffic} = N_{\text{allTag}} * (L_{\text{TagID}} + 2 * L) = \frac{n * (96 + 2 * L)}{\eta} \quad (25)$$

图 5 表明,当  $L=12$  时系统的通信量可达到

最小<sup>[13]</sup>。



(a) 标签数 4 000 到 5 000 段通信量曲线



(b) 标签数 6 000 到 6 500 段通信量曲线

图5 SBFSA 算法在不同位隙长度下的通信量曲线

### 4 算法描述

文中提出的 SBFSA 算法在保持了高吞吐率的同时,兼顾了通信量。以下为该算法的具体执行流程。

(1) 阅读器估计查询范围内的标签数量并发出一个 Query 查询命令。该 Query 命令包含一个  $Q$  值,  $Q$  取值依据为:若标签数大于 490,则取 8,否则取 5。数据帧长为  $2^Q$ 。

(2) 标签随机选择时隙,将其装入自己的槽计数器,同时生成两个 12 以内的随机数  $S_1$  与  $S_2$ 。

(3) 槽计数器为 0 的标签立即启动响应。该标签发送自己的 ID 信息并附加上一个 12 位宽的第一段位隙信息 FBS 和一个 12 位宽的第二段位隙信息 SBS。其中 FBS 中的第  $S_1$  位为 1,即  $\text{FBS}[S_1] = 1$ ,其他所有位均为 0。第二段位隙信息也是一样。

(4) 阅读器检查标签的返回 ID 信息与位隙信息。如果本时隙没有响应信号或者有无冲突的标签响应信号,则阅读器做相应的处理,然后转(8)。如果标签返回的 ID 信息有冲突,则转(5)。

(5) 阅读器查看 FBS 信息里有几位为 1。如果有  $n$  位为 1,则说明本时隙内至少有  $n$  个标签产生碰撞。阅读器按照从高位到低位的顺序,分别提取出含有 1 的位置信息,并将该信息包含到一个新的查询命令



QueryRepFBS 中。本时隙内的标签在收到该命令后,提取出该位置信息,并与自己的第一段位隙信息进行匹配。匹配成功的标签将自己的数据包返回给阅读器。阅读器检测返回的 ID 信息。如果没有冲突,就利用 ACK 命令进行确认;如果有冲突,则转(6)。

(6)阅读器查看整个位隙信息里有几位为 1,并按照从高位到低位的方式,分别提取出含有 1 的位置信息,将其包含到另一个新的查询命令 QueryRepSBS 中。本时隙内的标签在收到该命令后,提取出该位置信息,并与自己的整个位隙位进行匹配。匹配成功的再次返回自己的 ID 信息。阅读器查看是否有冲突;如果没有冲突,利用 ACK 命令确认;如果还有冲突,则丢包,进行 NAK 确认<sup>[14-15]</sup>。

(7)阅读器将 FBS 中含有 1 的最高位清零,如果 FBS 不为 0,则进入次高位,然后执行(5)-(7),直到 FBS 为 0。

(8)阅读器发送 QueryRep 命令,查询范围内的标签接到命令后将自己的槽计数器减 1,然后转(3)。如果本时隙是最后一个时隙,结束本轮查询并转(1)。

(9)某一轮查询中,所有时隙都是空时隙,整个查询过程结束。

## 5 结束语

RFID 是物联网发展的核心部分,它的发展将会给人类的生产、生活带来翻天覆地的变化。其巨大潜力将随着物联网的发展得到进一步的体现。防碰撞问题是 RFID 的一个关键问题。这将推动越来越多的研究人员投入到 RFID 防碰撞算法的研究中去。

文中理论推导得出:通过位隙分段,可以在保持高吞吐率的同时减少系统的总通信量。在此基础上,提出的分段式位隙分组帧时隙 Aloha 算法在阅读器查询范围标签数量较多时,仍然能够保持稳定且高效的信道利用率。

考虑到在确定性防碰撞算法中,可以利用发生碰撞标签的 ID 段进行再识别或进行通信量控制<sup>[16]</sup>。因此,在后续工作中,可以从以下两方面开展进一步的研究。

(1)在 SGBFSA 算法的基础上,将碰撞时隙中标签的 ID 段加以利用,进一步提高算法性能。

(2)将位隙段分三段以上,研究算法性能。由特殊到一般,探寻是否存在一般规律。

## 参考文献:

[1] Wang Yuhao, Liu Yi, Leung H, et al. A segment collision in-

version protocol for RFID tag reading[J]. IEEE Communications Letters, 2013, 17(10): 2008-2011.

- [2] Liu Xiaohui, Qian Zhihong, Zhao Yanhang, et al. An adaptive tag anti-collision protocol in RFID wireless systems[J]. China Communications, 2014, 11(7): 117-127.
- [3] Park J, Chung M Y, Lee T J. Identification of RFID tags in framed-slotted ALOHA with robust estimation and binary selection[J]. IEEE Communications Letters, 2007, 11(5): 452-454.
- [4] Hakeem M J, Raahemifar K, Khan G N. Novel modulo based ALOHA anti-collision algorithm for RFID systems[C]//Proc of IEEE international conference on RFID. [s. l.]: IEEE, 2014: 97-102.
- [5] Shih D H, Sun Poling, Yen D C, et al. Taxonomy and survey of RFID anti-collision protocols[J]. Computer Communications, 2006, 29(11): 2150-2166.
- [6] 黄玉兰. 物联网射频识别(RFID)技术与应用[M]. 北京:人民邮电出版社, 2013.
- [7] 王 钰. RFID 防碰撞算法研究[D]. 南京:南京邮电大学, 2011.
- [8] 王玉琢. UHF RFID 系统防碰撞研究[D]. 北京:北京邮电大学, 2011.
- [9] Deng D J, Tsao H W. Optimal dynamic framed slotted ALOHA based anti-collision algorithm for RFID systems[J]. Wireless Personal Communications, 2011, 59(1): 109-122.
- [10] 王 宇. 基于改进的 RFID 标签防碰撞技术的研究[D]. 长春:吉林大学, 2013.
- [11] 栗 华. UHF RFID 多标签防碰撞算法的研究与性能分析[D]. 济南:山东大学, 2011.
- [12] Wang Hui, Xiao Shengliang, Lin Feiyu, et al. Group improved enhanced dynamic frame slotted ALOHA anti-collision algorithm[J]. Journal of Supercomputing, 2014, 69(3): 1235-1253.
- [13] 张志涌, 杨祖樱. MATLAB 教程: R2010a[M]. 北京:北京航空航天大学出版社, 2011.
- [14] Li Xiaowu, Feng Quanyuan. An improved dynamic framed slotted ALOHA using posteriori probability tag estimation algorithm in the mobile radio frequency identification systems[J]. Journal of Computational and Theoretical Nanoscience, 2014, 11(6): 1417-1421.
- [15] Dhakal S, Shin S. Precise-optimal frame length based collision reduction schemes for frame slotted aloha RFID systems[J]. KSII Transactions on Internet and Information Systems, 2014, 8(1): 165-182.
- [16] Lai Yuancheng, Hsiao Ling-Yen, Lin Bor-Shen. An RFID anti-collision algorithm with dynamic condensation and ordering binary tree[J]. Computer Communications, 2013, 36(17-18): 1754-1767.